Amendments to the Claims

Please amend Claims 1, 2, 12-14, and 17-22. The Claim Listing below will replace all prior versions of the claims in the application:

Claim Listing

1. (Currently amended) An agent process for controlling access to digital assets in a network of data processing devices, the process comprising:

defining a point-of-use security perimeter that includes the operating system kernels of two or more data processing devices;

defining one or more policy violation predicates that serve to implement policy logic and that are asserted at the point-of-use of a digital asset upon an occurrence of a possible risk of use, outside of the security perimeter, of [[a]] the digital asset by an end user ~~outside of the security perimeter~~;

sensing atomic events [[from]] within an operating system kernel of a user client device, the atomic events being low level kernel events and being sensed upon actions relating to authorized access to a digital asset by the end user of the user client device;

aggregating multiple atomic level events to determine a combined event; and

asserting a policy violation predicate upon an occurrence of a combined event that violates a predefined digital asset usage policy that indicates a risk of use of the digital asset outside of the security perimeter.

2. (Currently amended) A process as in Claim 1 wherein the step of asserting the policy violation predicate is implemented in [[an]] the operating system kernel of the client user device.

3. (Original) A process as in Claim 1 additionally comprising:

preventing a user from accessing the digital asset if the policy predicate indicates a violated policy.

4. (Original) A process as in Claim 3 wherein the preventing step includes an IRP intercept.

5. (Original) A process as in Claim 1 wherein the combined event is a time sequence of multiple atomic level events.

6. (Original) A process as in Claim 1 additionally comprising:

prompting a user to document a reason for a policy violation, prior to granting access to the digital asset.

7. (Previously presented) A process as in Claim 1 additionally comprising:

asserting multiple policy violation predicates prior to indicating a risk of use of the digital asset outside of the security perimeter.

8. (Original) A process as in Claim 2 that operates independently of application software.

9. (Original) A process as in Claim 1 additionally comprising:

notifying a user of a policy violation, and then permitting access to the digital asset.

10. (Original) A process as in Claim 2 wherein the sensors, aggregators, and asserting steps operate in real time.

11. (Original) A process as in Claim 1 additionally comprising:

determining the identity of a particular file in the asset access event.

12. (Currently amended) A system for controlling access to digital assets in a network of data processing devices, the system comprising:

a digital asset usage policy server storing one or more digital asset usage policies ~~configured~~ programmed to be applied to a point-of-use security perimeter, the security

perimeter comprising the operating system kernels of two or more data processing devices;

an atomic event sensor, the sensor located within an operating system kernel within an end user client device and ~~configured~~ programmed to sense atomic events [[from]] within the operating system kernel, the atomic events being low level kernel events and being sensed by the sensor upon actions relating to authorized access to one or more digital assets by an end user of the end user client device;

an atomic level event aggregator ~~configured~~ programmed to determine the occurrence of an aggregate event that comprises more than one atomic level asset access event; and

a policy violation detector ~~configured~~ programmed to determine whether an aggregate event has occurred that violates a predefined digital asset usage policy that indicates a risk of use of a digital asset outside the security perimeter.

13. (Currently amended) A system as in Claim 12 wherein the policy violation detector is located in [[an]] the operating system kernel of the user client device.

14. (Currently amended) A system as in Claim 12 wherein the policy violation detector is ~~configured~~ programmed to determine a violated policy type.

15. (Previously presented) A system as in Claim 14 wherein the policy violation detector includes an IRP intercept.

16. (Previously presented) A system as in Claim 12 wherein the combined event is a time sequence of multiple atomic level events.

17. (Currently amended) A system as in Claim 12 further including a user interface within the client device ~~configured~~ programmed to require the end user to document a reason for a policy violation prior to granting access to the digital asset.

18. (Currently amended) A system as in Claim 12 wherein the policy violation detector is additionally ~~configured~~ programmed to assert multiple policy violation predicates prior to indicating a risk of use of the digital asset outside of the security perimeter.

19. (Currently amended) A system as in Claim 13 that is ~~configured~~ programmed to operate independently of application software.

20. (Currently amended) A system as in Claim 12 wherein the user client device includes a user interface ~~configured~~ programmed to notify the end user of a policy violation and to permit access to the digital asset once a reason for the violation is provided by the end user.

21. (Currently amended) A system as in Claim 12 wherein the sensor, aggregator and detector are ~~configured~~ programmed to operate in real time.

22. (Currently amended) A system as in Claim 12 wherein the detector is additionally ~~configured~~ programmed to determine the identity of a particular file in the atomic level asset event.